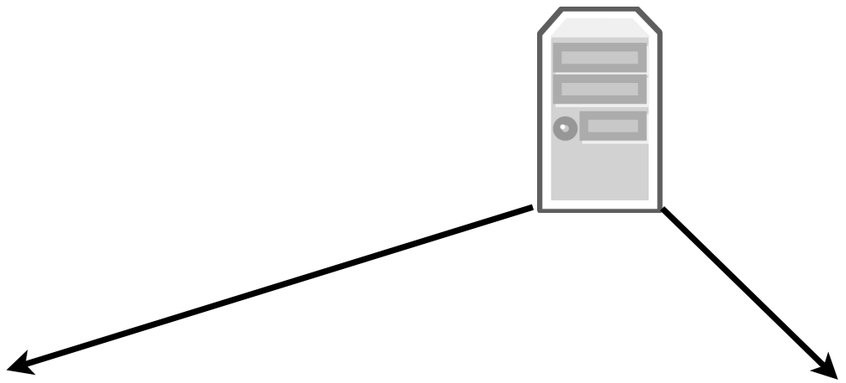
Address Resolution Protocol (ARP)

Layers of Addresses

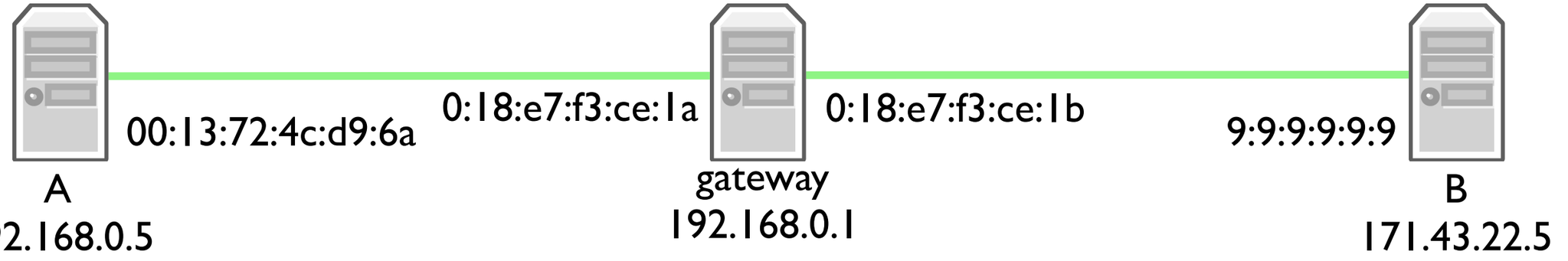
Application
Presentation
Session
Transport
Network
Link
Physical



Host: 171.67.76.65

Interface: 00:13:72:4c:d9:6a

Addressing Problem



Source

destination

netmask: 128.0.0.0

192.168.0.16 128.0.0.0
= 128.0.0.0

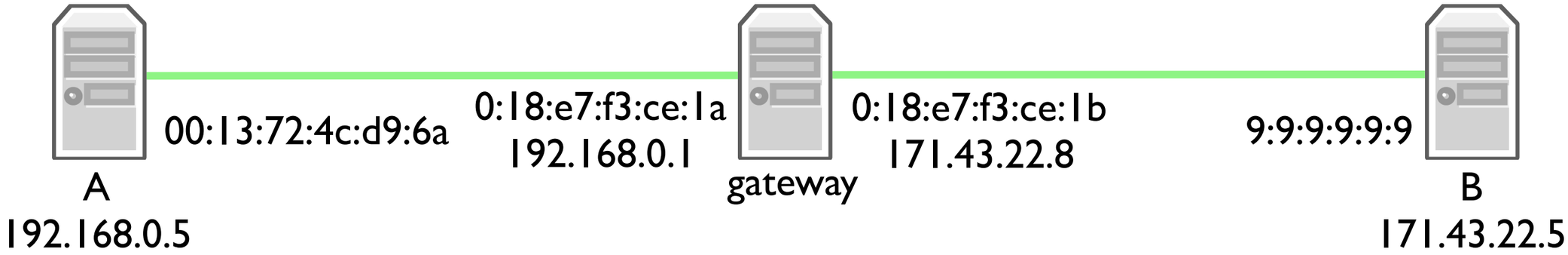
171.43.22.5 & 128.0.0.0
= 128.0.0.0

for gateway to be in same network as B, netmask must be 128.0.0.0

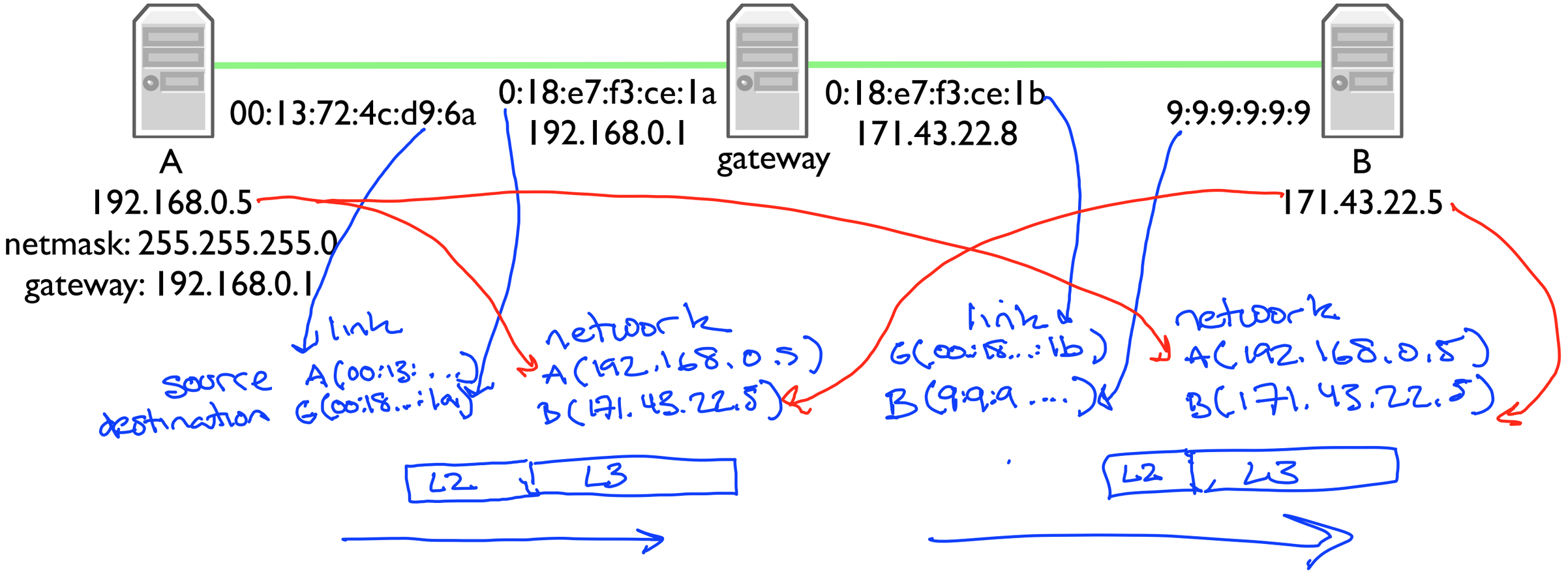
but this means 192.168.0.5 is in the same network too!

✗ 192.168.0.5 & 128.0.0.0
= 128.0.0.0

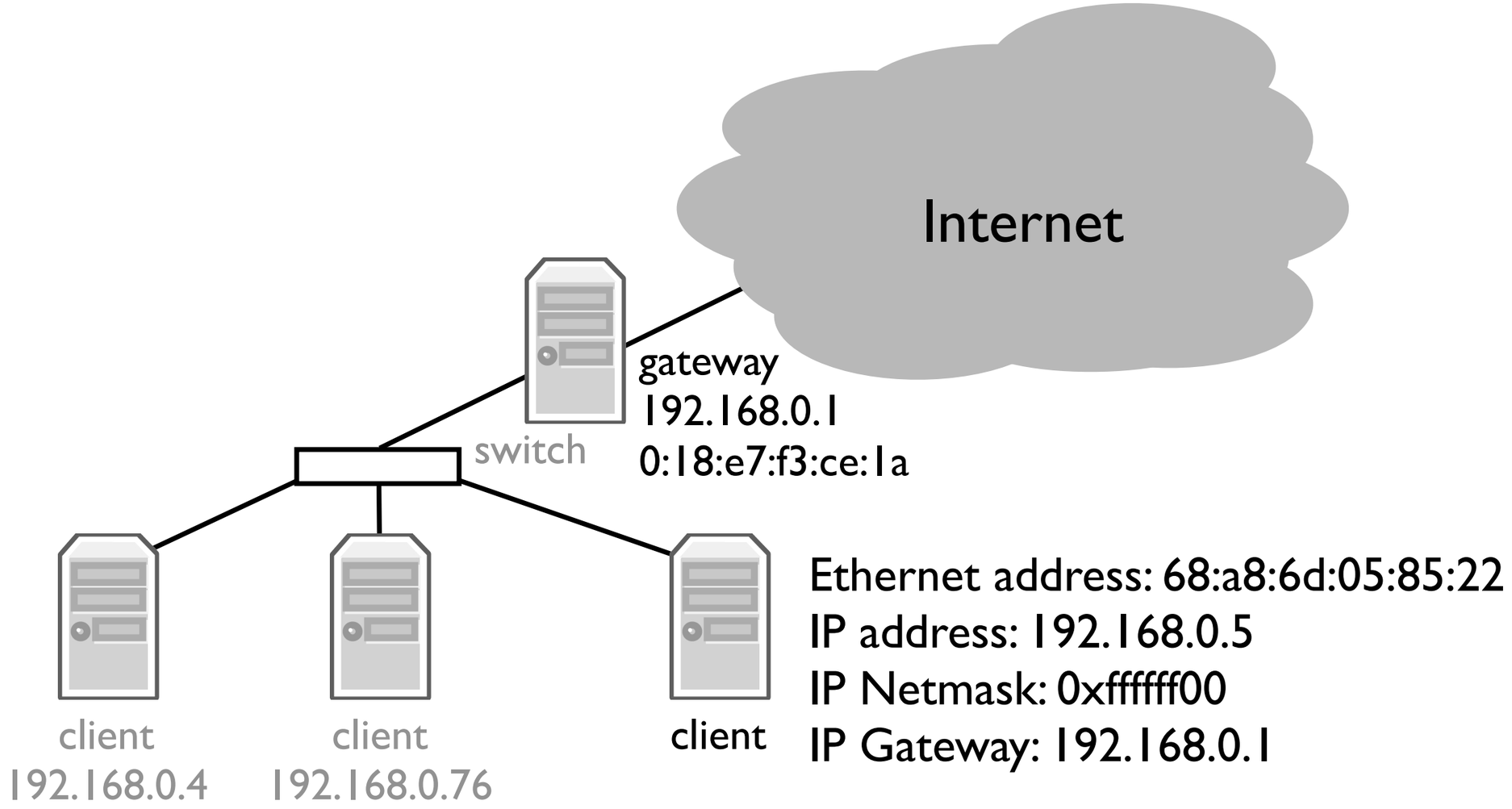
Example Addressing



Encapsulation



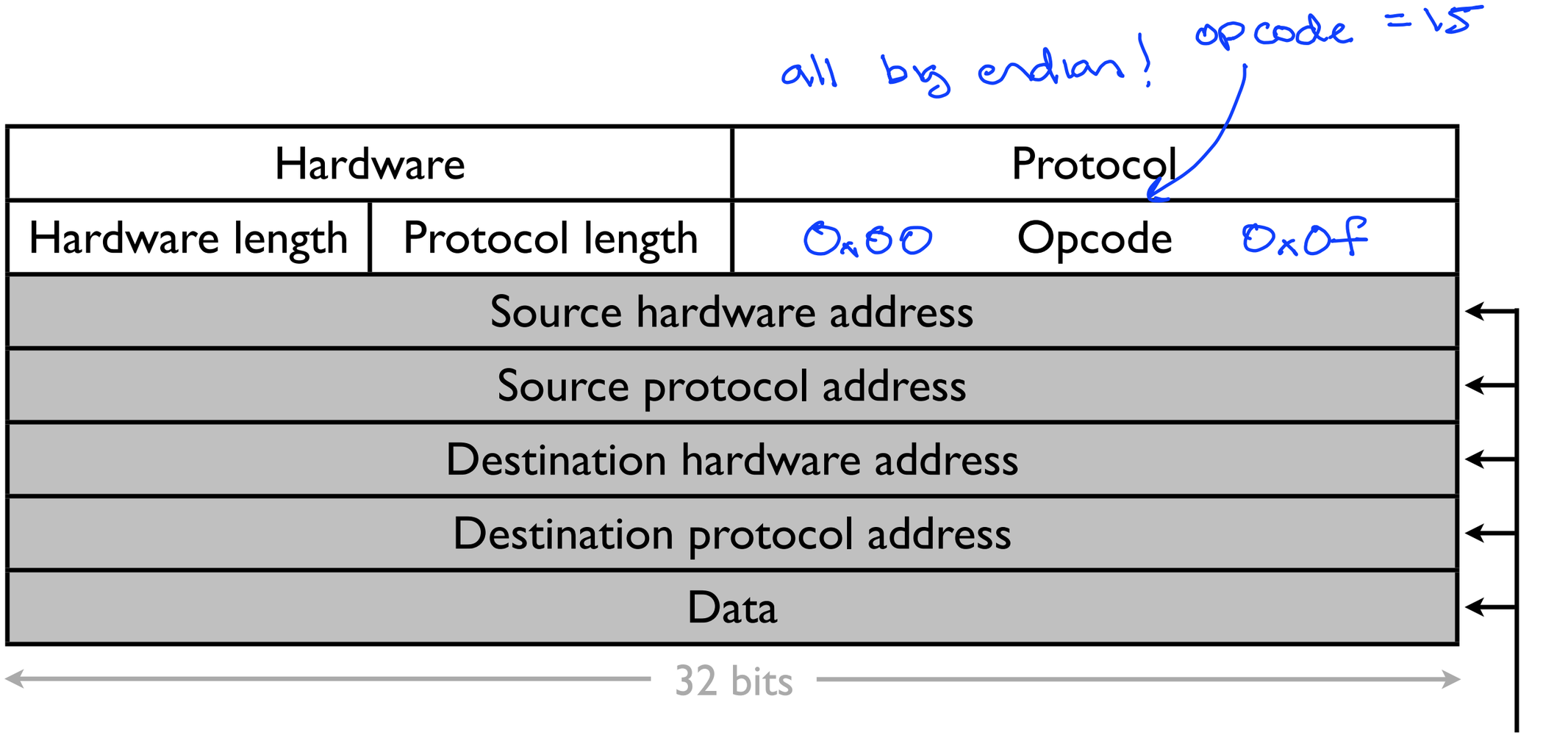
Example Problem



Address Resolution Protocol

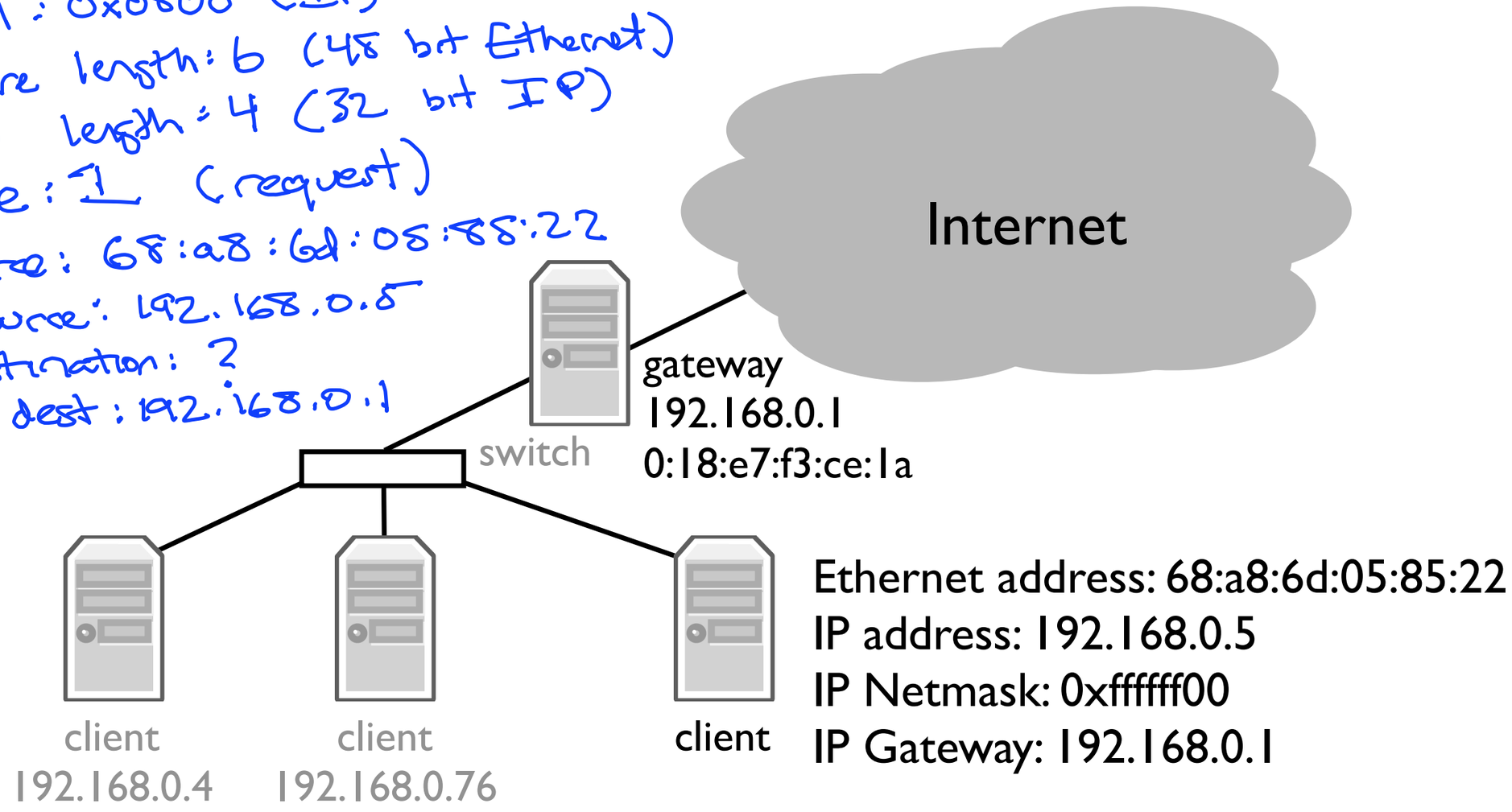
- Generates mappings between layer 2 and layer 3 addresses
  - Nodes cache mappings, cache entries expire
- Simple request-reply protocol
  - "Who has network address X?"
  - "I have network address X."
- Request sent to link layer broadcast address
- Reply sent to requesting address (not broadcast)
- Packet format includes redundant data
  - Request has sufficient information to generate a mapping
  - Makes debugging much simpler
- No "sharing" of state: bad state will die eventually

ARP Packet Format (RFC826)



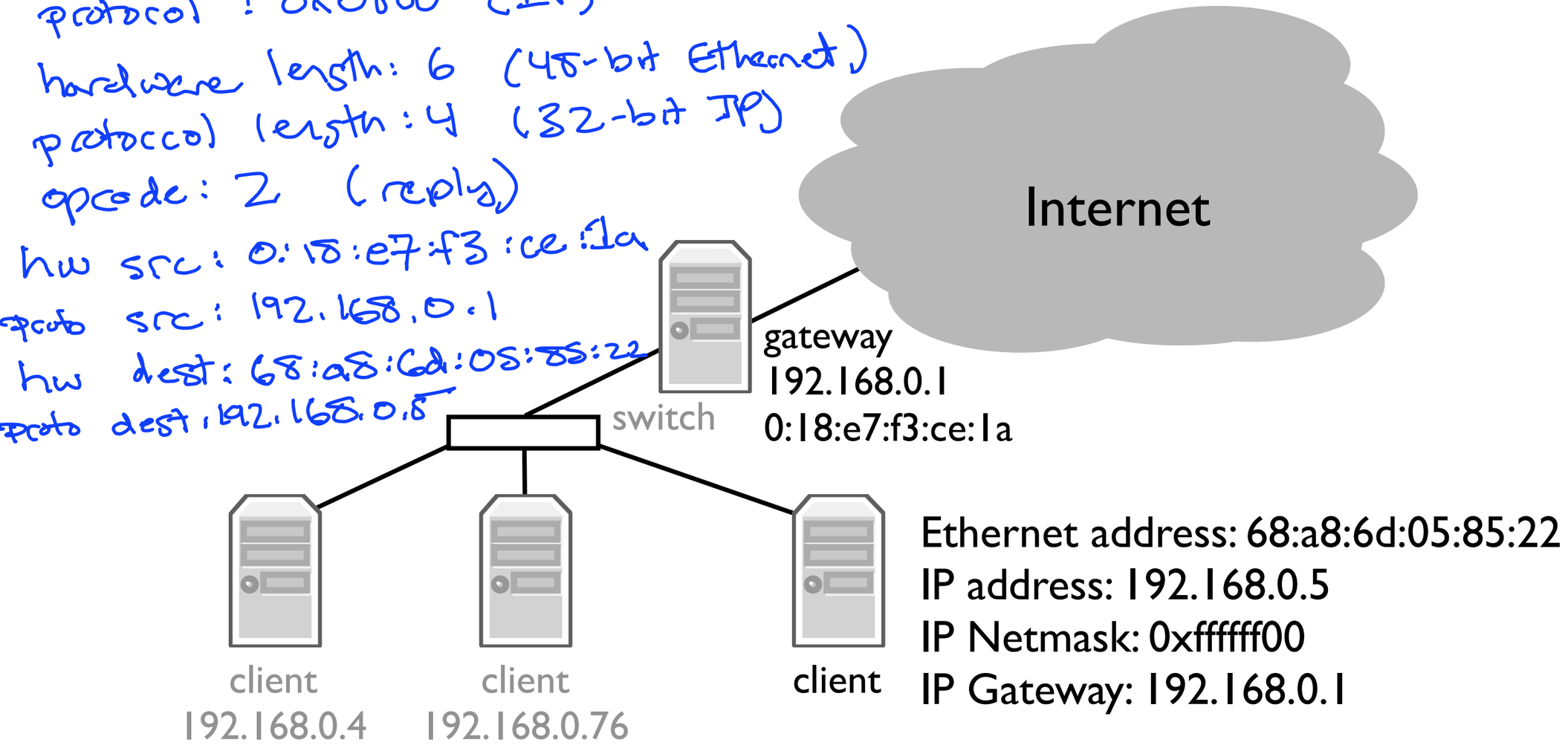
source: 68:a8:6d:05:85:22
dest: broadcast (ff:ff:ff:ff:ff:ff)
hardware: 1 (Ethernet)
protocol: 0x0800 (IP)
hardware length: 6 (48 bit Ethernet)
protocol length: 4 (32 bit IP)
opcode: 1 (request)
HW source: 68:a8:6d:05:85:22
protocol source: 192.168.0.5
HW destination: ?
protocol dest: 192.168.0.1

ARP Request



ARP Reply

hardware: 1 (Ethernet)
protocol : 0x0800 (IP)
hardware length: 6 (48-bit Ethernet)
protocol length: 4 (32-bit IP)
opcode: 2 (reply)
hw src: 0:18:e7:f3:ce:1a
proto src: 192.168.0.1
hw dest: 68:a8:6d:05:85:22
proto dest: 192.168.0.5



Ethernet address: 68:a8:6d:05:85:22
IP address: 192.168.0.5
IP Netmask: 0xffffff00
IP Gateway: 192.168.0.1